DatahubIO

Dokumentacja aplikacji

Autorzy:

Bartłomiej Zapart  
Dawid Pastuszka  
Jakub Janicki  
Maciej Kazalski  
Michał Stencel

# Plik konfiguracyjny

Plik formatu .json o określonej strukturze.

Składa się z trzech zagnieżdżonych warstw:

* Grupy (zbioru wykresów),
* Wykresu (zbioru serii danych wyświetlanych na wspólnej przestrzeni),
* Danych (konkretna seria danych).

Poniżej opisano jak powinien wyglądać takowy plik: na niebiesko oznaczono nazwy wymaganych pól, natomiast w nawiasach <> opis i typ zawartości:

*"nazwa\_pola"*: <opis:**typ**>,

*"nazwa\_pola"*: <opis:”konkretna\_wartość\_możliwa\_do\_wpisania”>,

*"nazwa\_pola"*: <opis:**typ**|”alternatywna\_wartość\_możliwa\_do\_wpisania”>,

W listach, napisy GRUPA, WYKRES i SPECYFIKACJA\_DANYCH oznaczają wyjaśnione dalej obiekty.

### Struktura ogólna

{

*"groups"*: [

GRUPA[1],

GRUPA[2],

…

GRUPA[N]

]

}

### Struktura Grupy

{

*"name"*: <nazwa\_grupy:**string**>,

*"description"*: <opis\_grupy:**string**>,

*"charts"*: [

WYKRES[1],  
 WYKRES[2],

…

WYKRES[M]

]

}

### Struktura wykresu

{

*"description"*: <opis:**string**>,

*"startDate"*: <data\_pierwszego\_pomiaru:**dateTimeString**>,

*"endDate"*: <data\_ostatniego\_pomiaru:**dateTimeString**|”PRESENT”>,

*"chartType"*: <typ\_grafu:”lineGraph”>,

*"title"*: <tytuł:**string**>,

*"xLabel"*: <nazwa\_osi\_OX:**string**>,

*"yLeftLabel"*: <nazwa\_lewej\_osi\_OY:**string**>,

*"yRightLabel"*: <nazwa\_prawej\_osi\_OX:**string**>,

*"xUnit"*: "date time", **<???>**

*"yUnit"*: "date time", **<???>**

*"leftAxis"*: {

*"unit"*: <jednostka\_lewej\_osi\_OY:**string**>,

*"decimals"*: <ilość\_miejsc\_po\_przecinku:**int**>

},

*"rightAxis"*: {

*"unit"*: <jednostka\_prawej\_osi\_OY:**string**>,

*"decimals"*: <ilość\_miejsc\_po\_przecinku:**int**>

},

*"timestampsPath"*: [

"timestamp" <???>

],

*"dataDetails"*: {

*"endpoint"*: <link\_do\_poboru\_danych:**string**>,

*"data"*: [

SPECYFIKACJA\_DANYCH[1],

SPECYFIKACJA\_DANYCH[2],

…

SPECYFIKACJADANYCH[K]

]

},

*"extraData"*: <dane dodatkowe> <???>

}

### Struktura danych

{

*"name"*: <nazwa\_serii\_danych:**string**>,

*"from"*: <ścieżka\_zagnieżdżenia\_danych:**list<string>**>,

*"axis"*: <przypisanie\_do\_osi\_danych:”left”|”right”>

}

Ścieżka zagnieżdżenia danych to lista z informacją w które pola zwracanego przez link pliku .json należy się kolejno zagłębić, aby uzyskać wartości do konkretnej serii danych, np.:  
  
dla pliku{

{

*"name"*: “somename”,

*"values"*: {

*"series\_a"*: {

*"example"*: 2.3

}

*"series\_b"*: {

*"something"*: 1.0

}

}

}

}  
aby uzyskać wartość 2.3 pola “example” należy podać następującą ścieżkę zagnieżdżenia:[

{

*"from"*: [

”values”,

”series\_a”,

”example”

]

}

# 

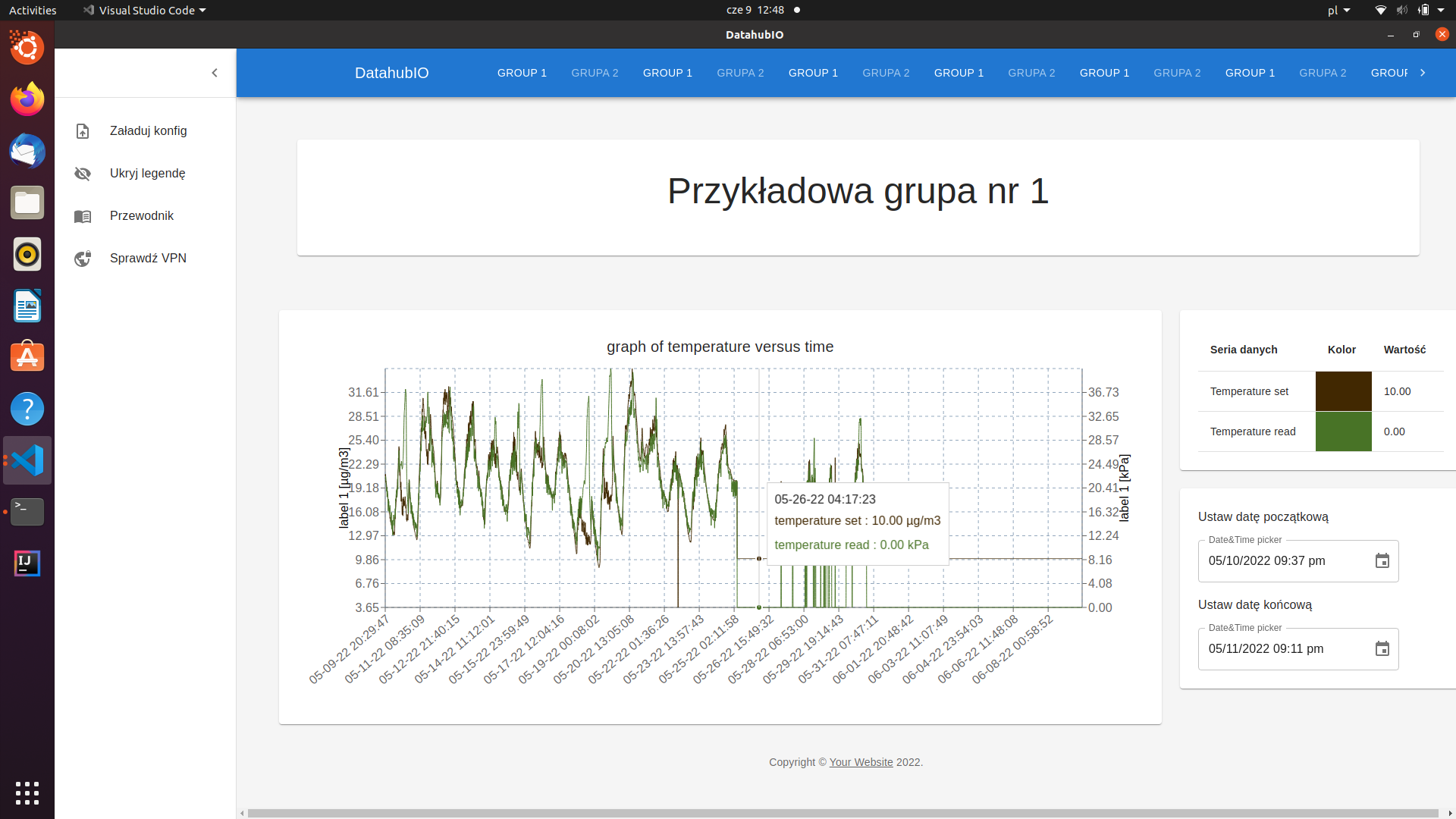
# Opis aplikacji

### Uruchomienie

Należy pobrać i uruchomić odpowiedni dla systemu (windows, linux lub mac) plik wykonywalny z folderu **exec**.  
  
W przypadku chęci zbudowania projektu własnoręcznie należy pobrać kod projektu, zainstalować **npm** oraz **yarn** i uruchomić następujące komendy:  
  
> **npm i**

> **npm run-script electron:package:<system: linux|windows|mac>**

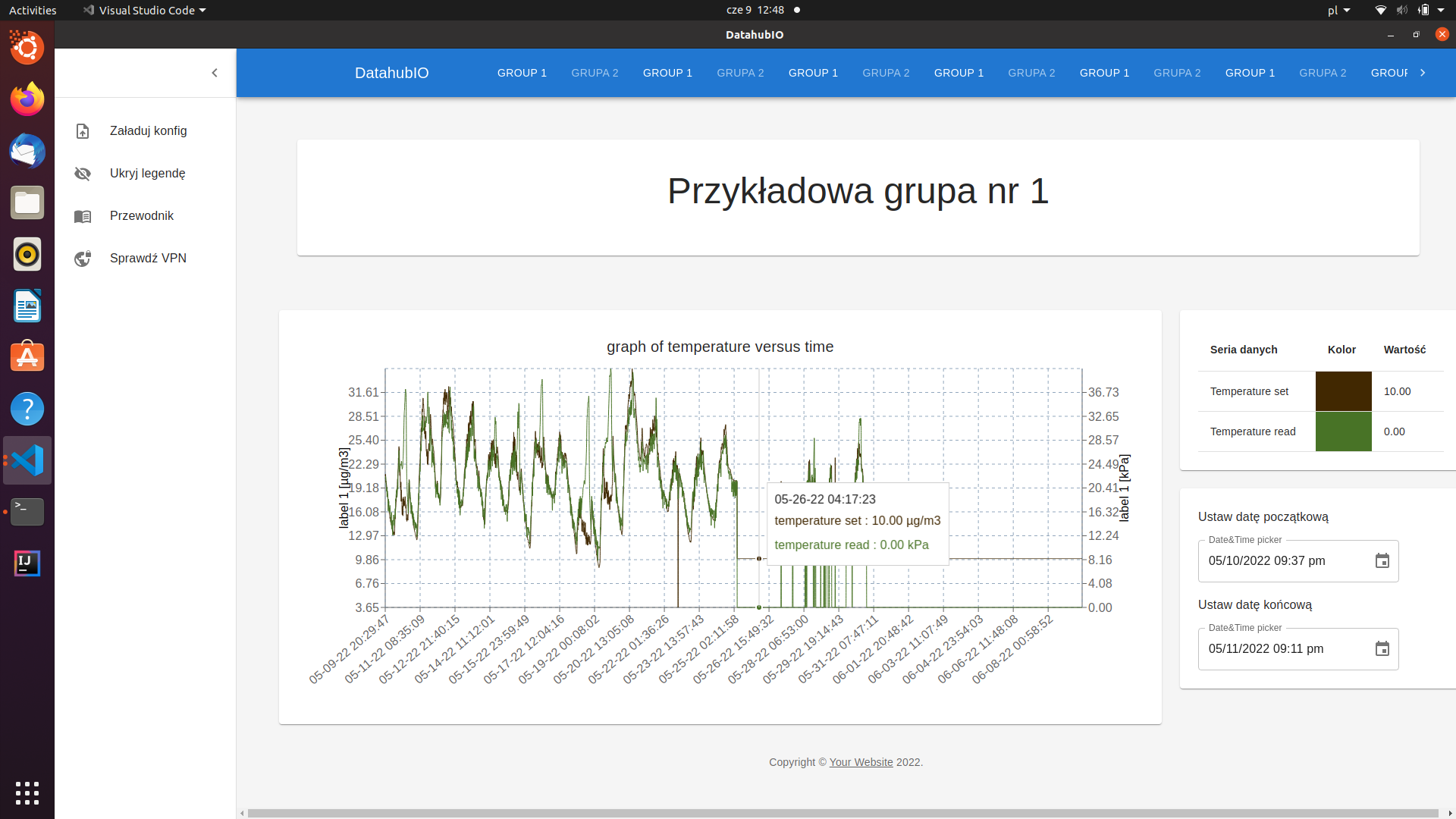
### Widok



#### Pasek opcji

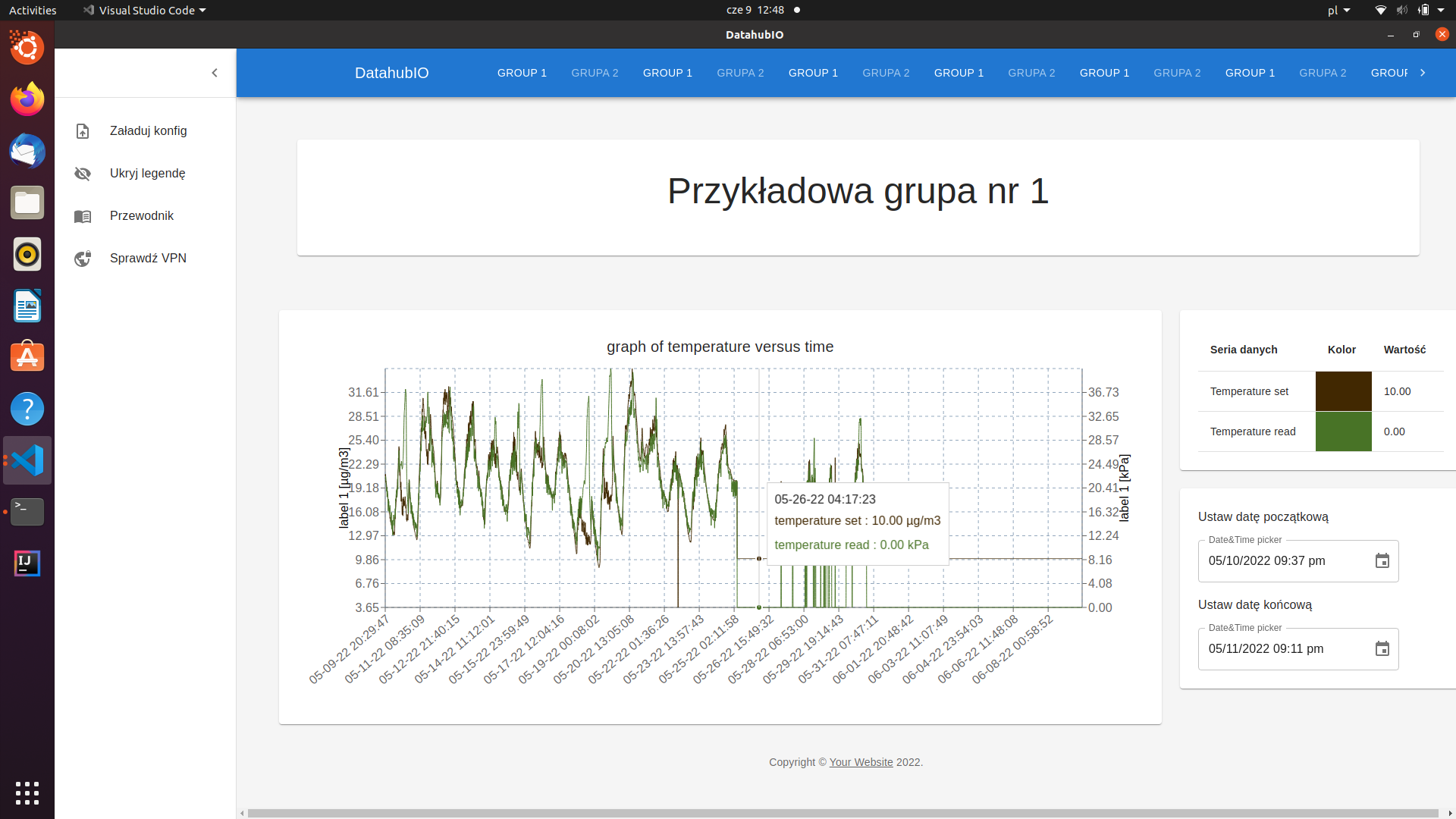
Udostępnia opcje:

1. załadowania nowego pliku konfiguracyjnego,
2. przełączanie widżetu legendy,
3. dostęp do przewodnika,
4. sprawdzenie stanu łącza VPN.



#### Pasek nawigacji grup

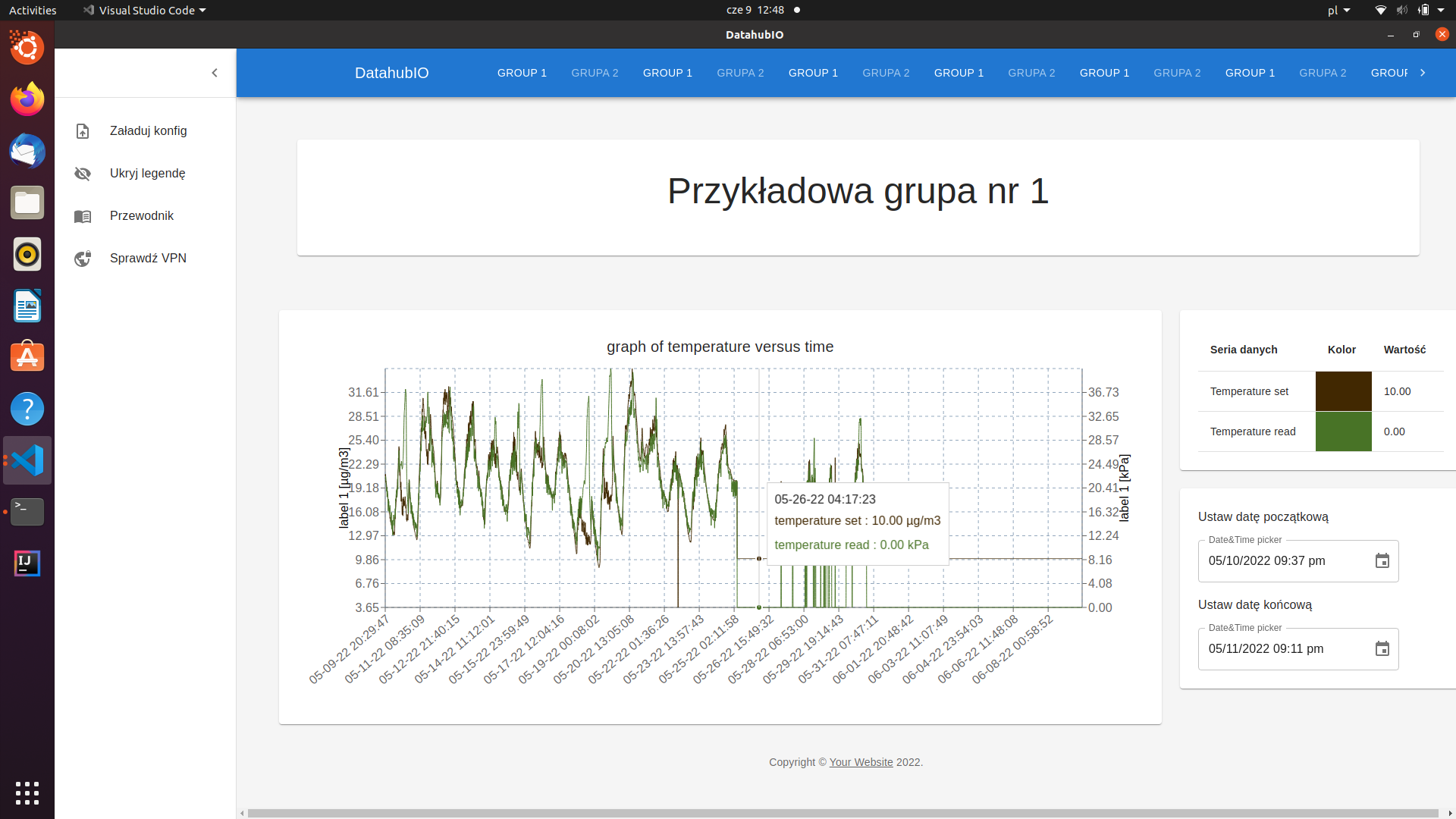
Umożliwia przełączanie się między grupami.

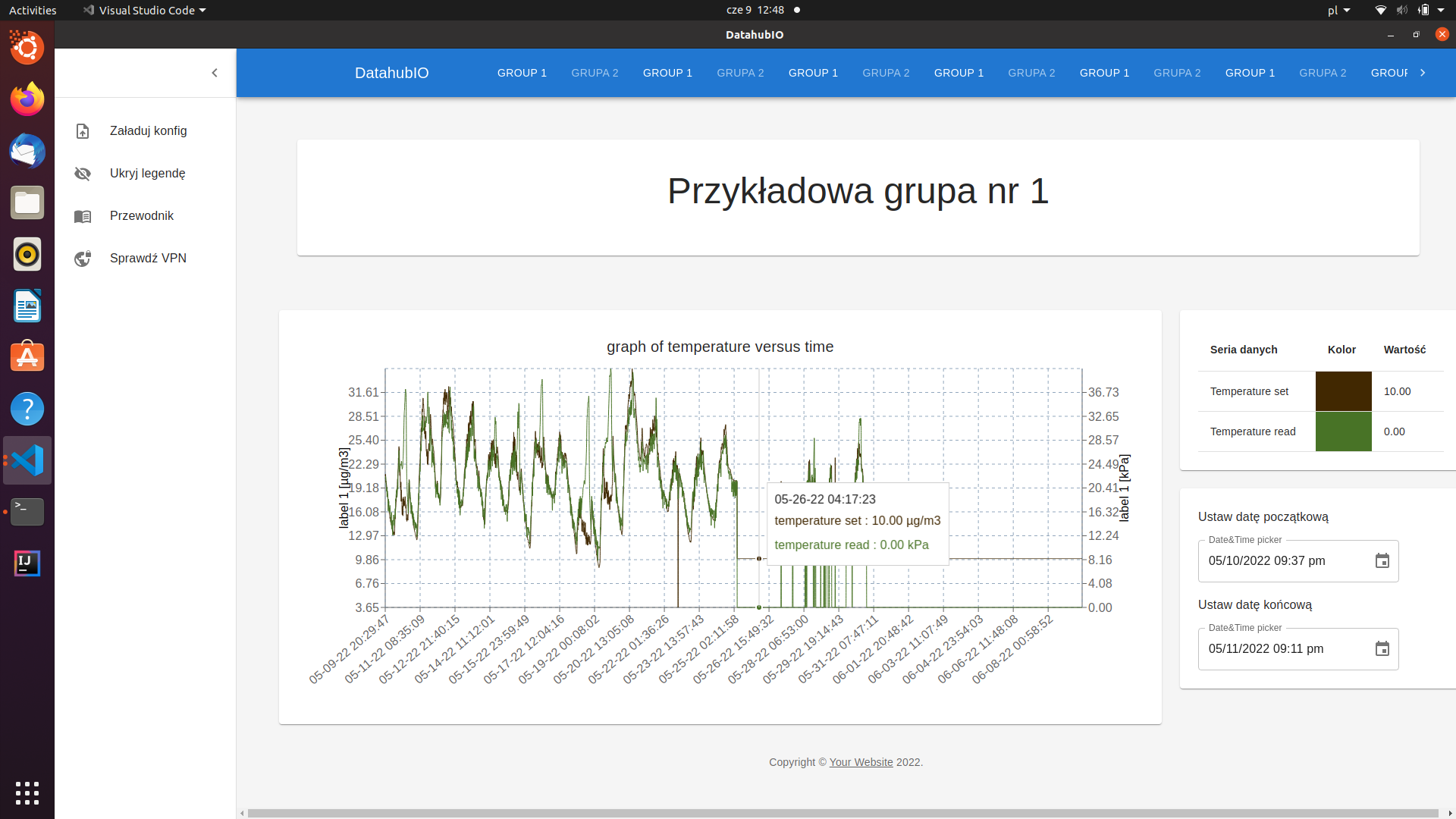


#### 

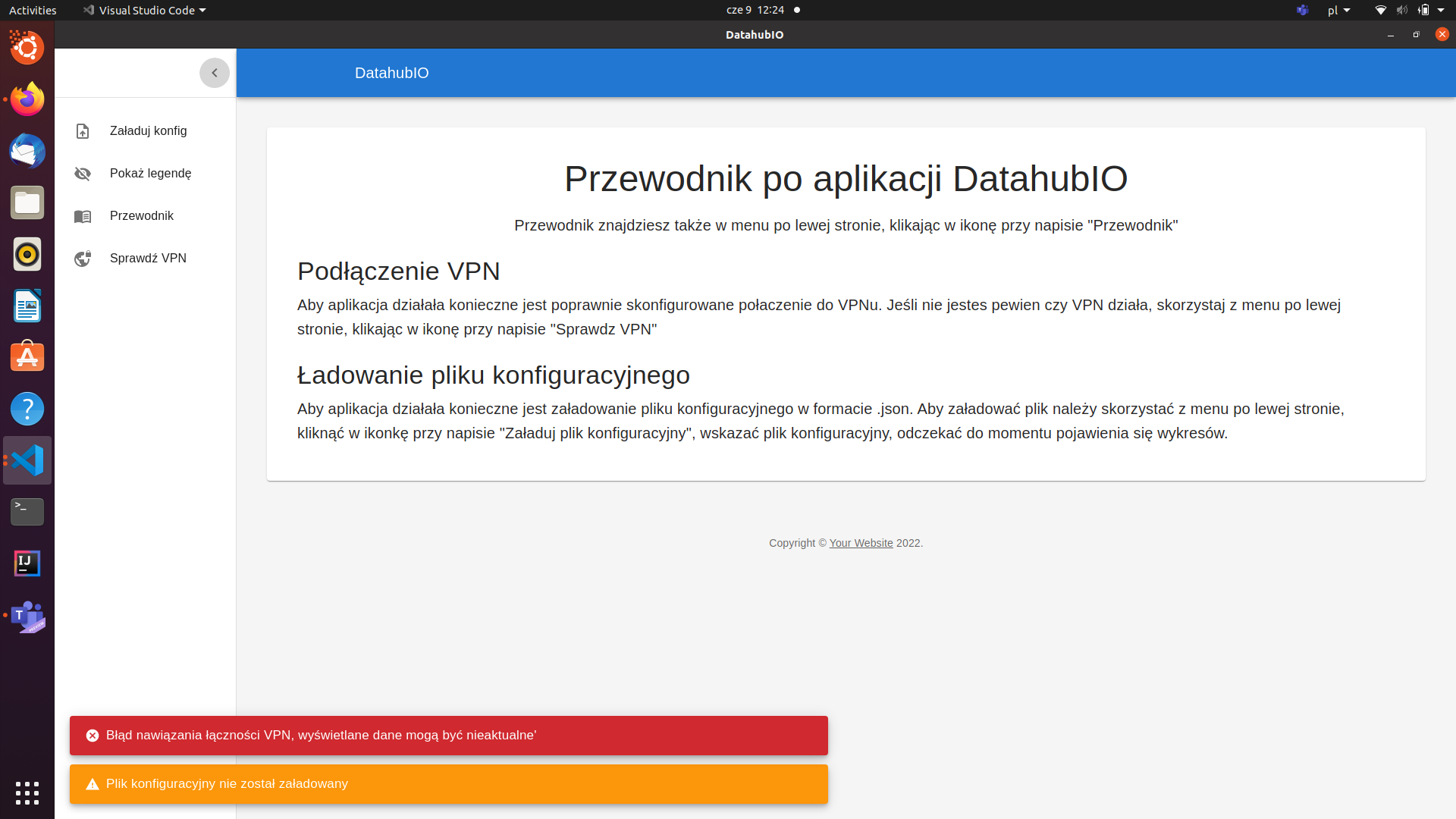
#### Wykresy

Wewnątrz grupy wyświetlone będą wykresy:

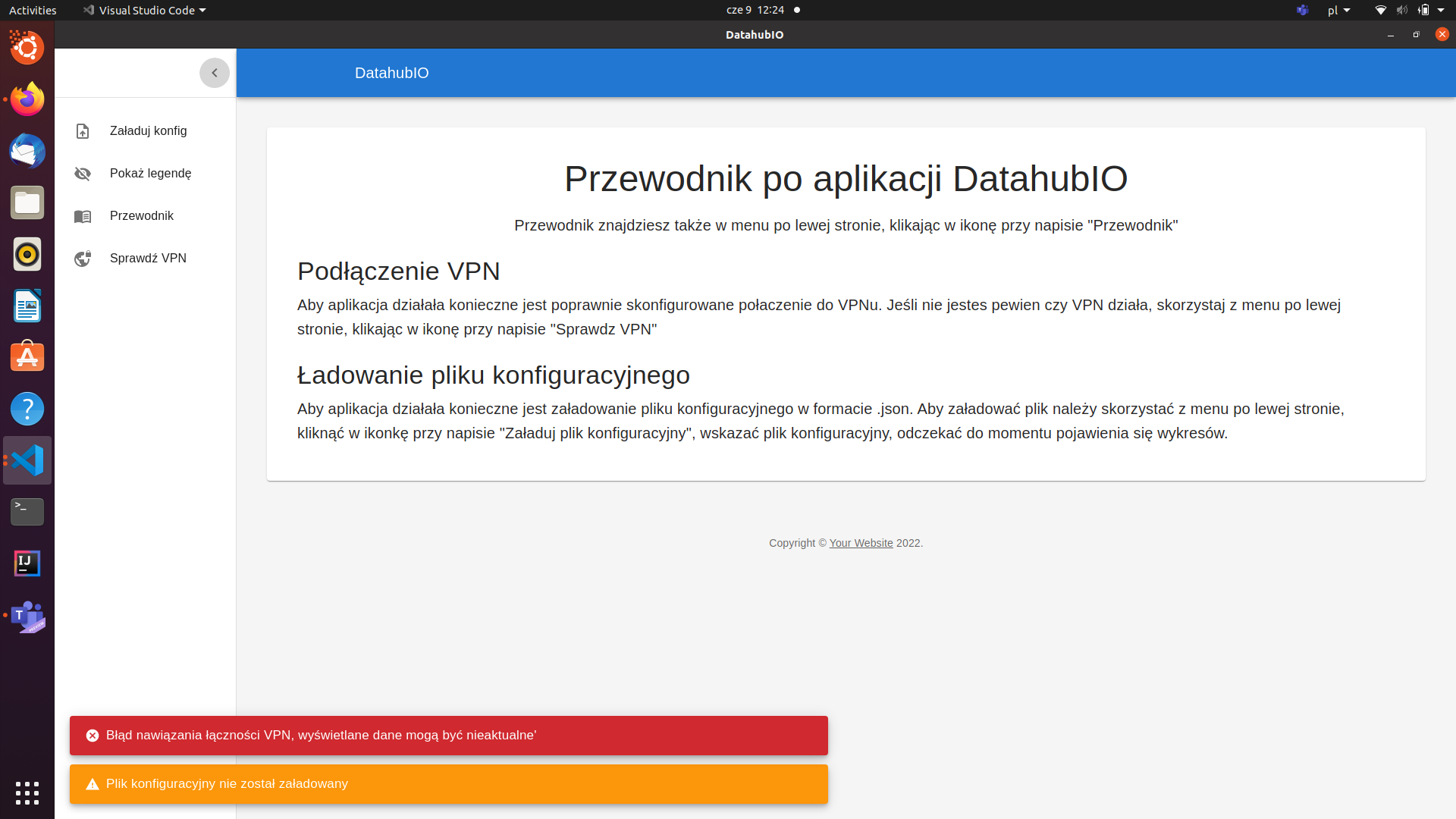
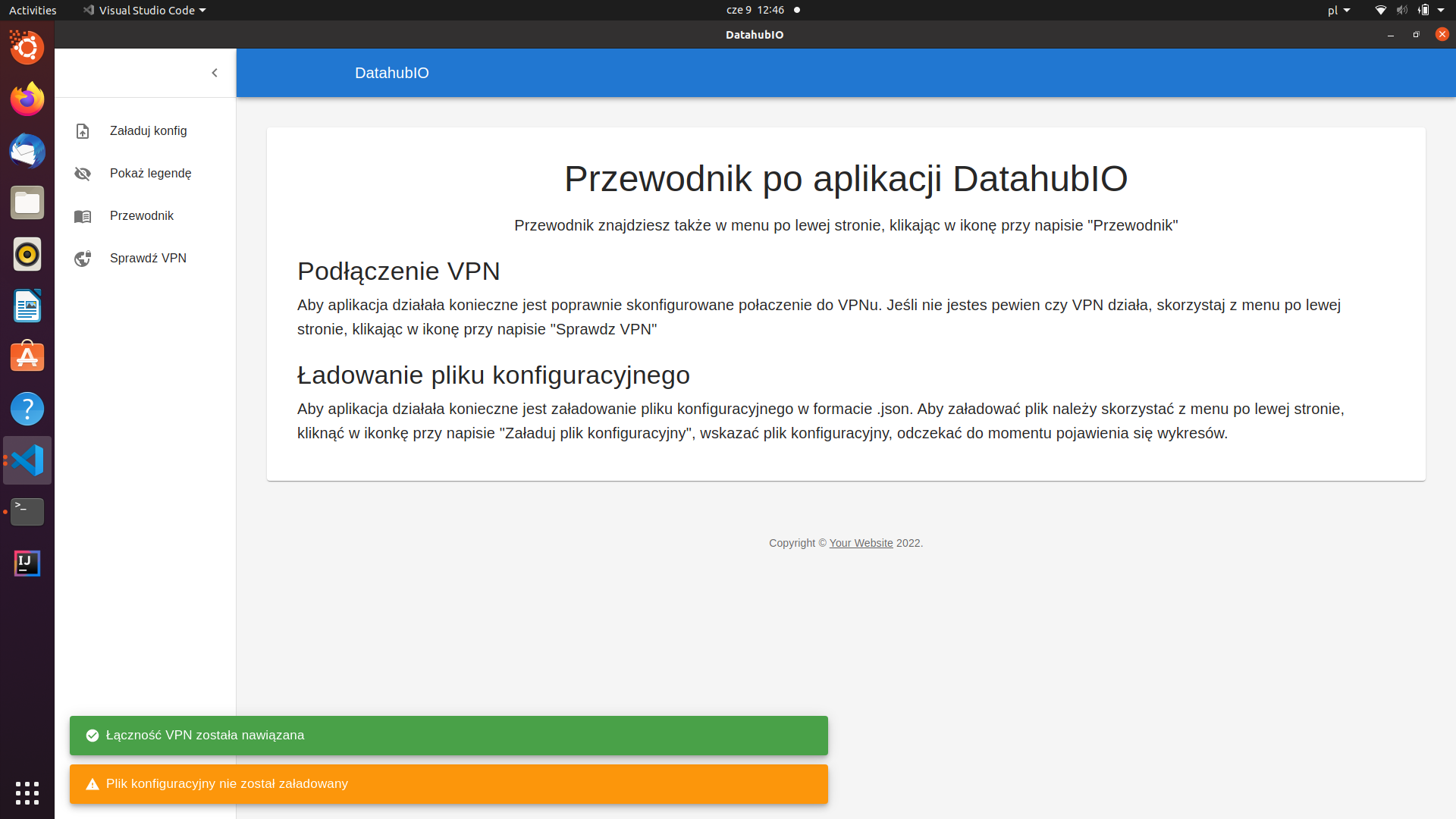
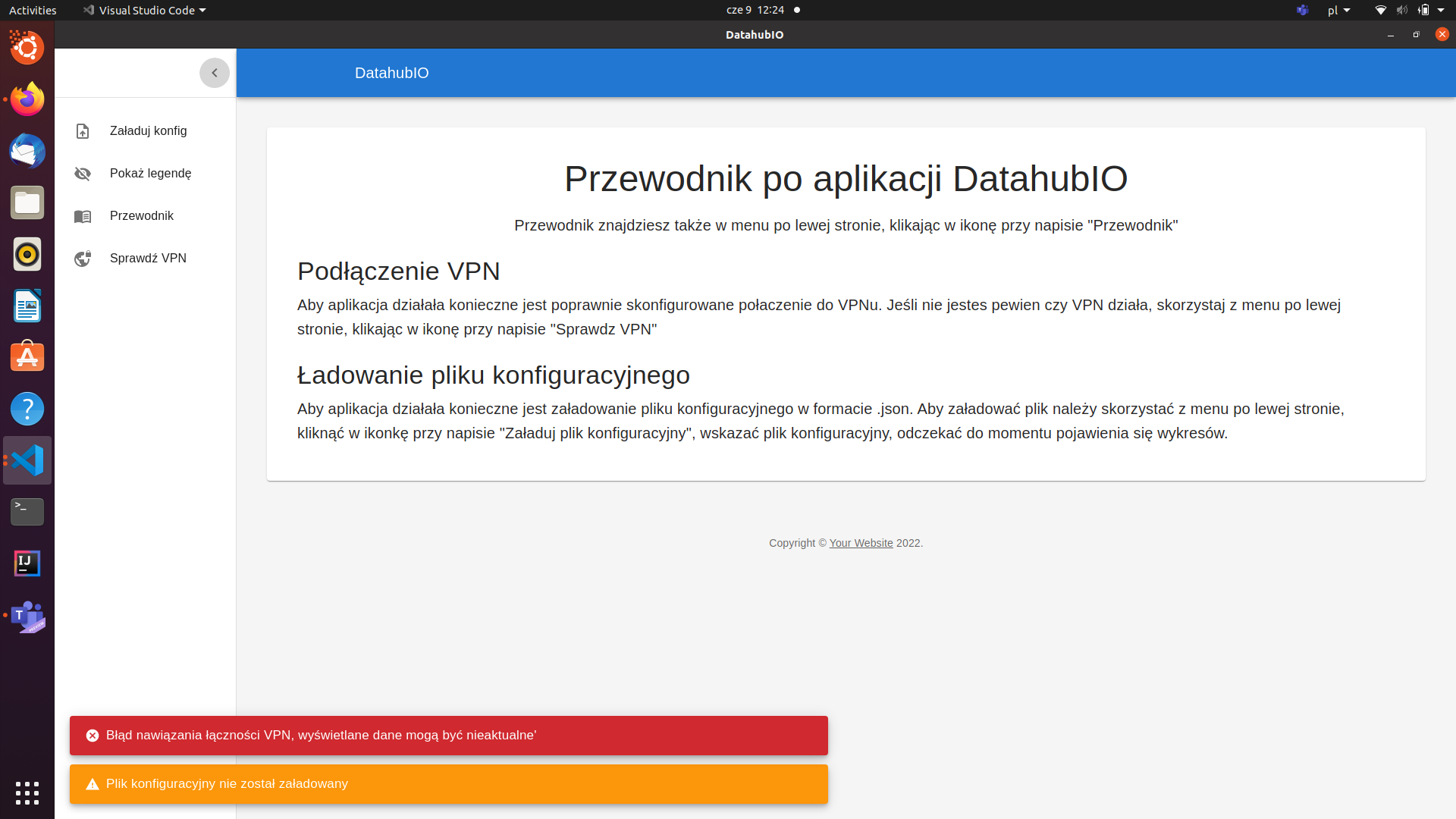
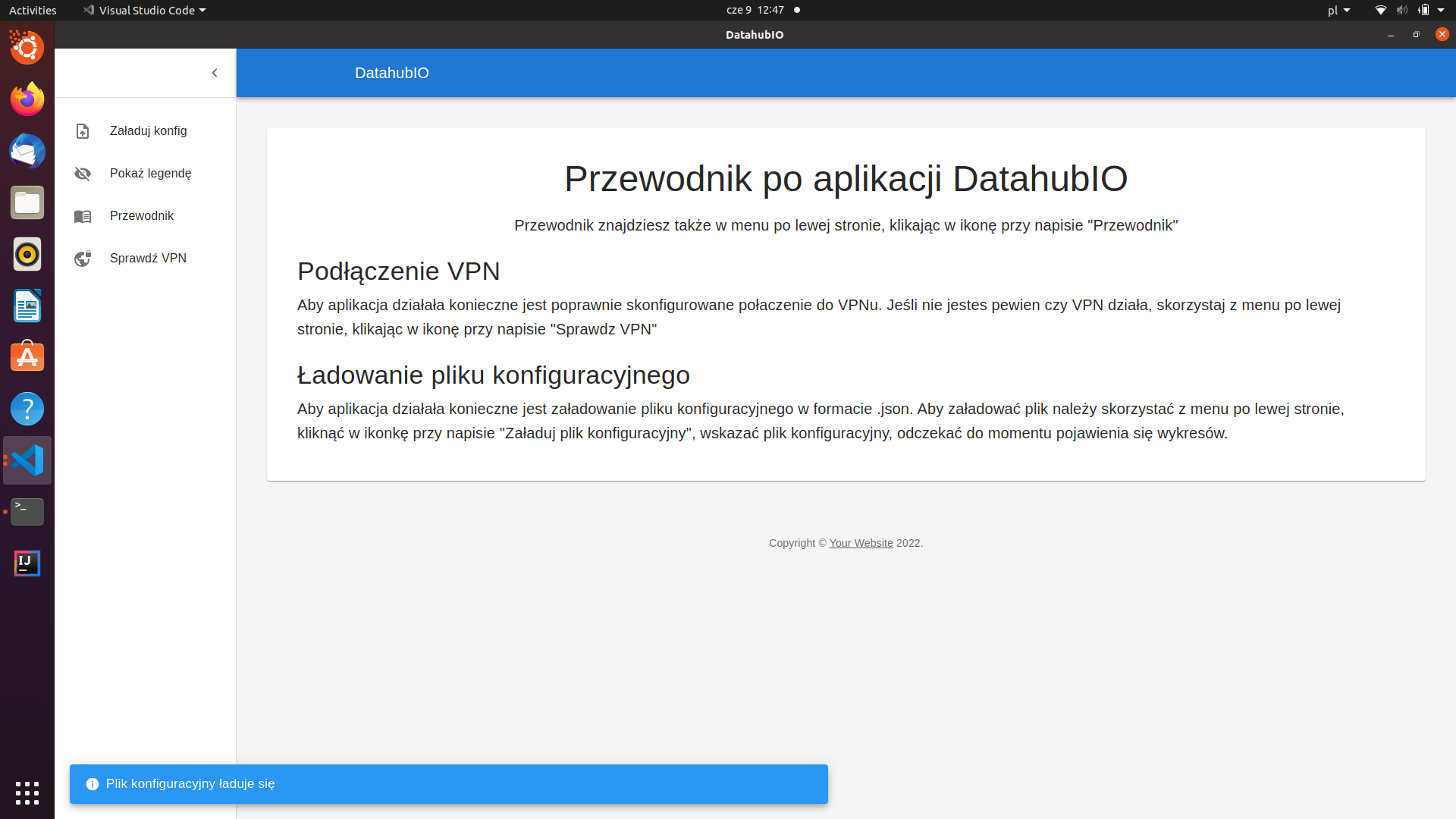


wraz z widrzetami (możliwymi do ukrycia przez pasek opcji) zawierającymi **legendę** oraz **sekcję zarządzania zakresem czasowym wykresu**.  


#### Komunikaty



Aplikacja wyświetla cztery rodzaje komunikatów:

* dwa dotyczące łącza VPN  
    
  
* dwa dotyczące pliku konfiguracyjnego  
    
  

#### Przewodnik

Możliwy do otwarcia poprzez pasek opcji. Udostępnia opis ułatwiający nawigację po aplikacji.  
